

REMARKS

Applicant thanks the examiner for her time during the June 4, 2007, telephone call in which the examiner confirmed that the initialed references listed on the PTO-1449 forms attached to the April 10, 2007, Office Action, were considered. The examiner further confirmed that the reference "Computer Architecture A quantitative Approach" to Patterson et al. (listed as reference "EF" in the attached initialed PTO-1449 form) was previously considered (as indicated in the initial PTO-1449 form attached to the November 28, 2005, Office Action), and therefore did not have to be considered again. The examiner further confirmed that there is no need to re-submit any of the PTO-1449 forms previously submitted.

Claims 17-21, 23-26 and 28-29 are pending in the above-referenced patent application. Claims 17 and 26 are independent.

Applicant amended independent claim 1 to clarify that the shifted value of the operand is shifted according to a shift control parameter specified by the local register instruction. Support for this clarification is provided, for example, at page 11, line 29, to page 12, line 31 of the application. Applicant similarly amended independent claim 26.

The examiner rejected claims 17, 20-21, 23-26 and 28-29 under 35 U.S.C. §103(a) as being unpatentable over U.S. Patent No. 6223,277 to Karguth, in view of the reference "Programming, Compiling and Executing Partially-Ordered Instruction Streams on Scalable Shared-Memory Multiprocessors" by D. K. Probst (hereinafter Probst.)

The examiner also rejected claims 18-19 under 35 U.S.C. §103(a) over Karguth, in view of Probst, and further in view of the reference Computer Systems Design and Architecture by V. P. Heuring and H. F. Jordan (hereinafter Heuring.)

Applicant's amended independent claim 17 recites "a local register instruction that loads one or more bytes, specified by a multiple-bit field of the instruction, within a local destination register with a shifted value of another operand, the field representing a mask in which each bit of the mask identifies a different byte of the destination register, the shifted value of the other operand being shifted according to a shift control parameter specified by the local register instruction."

Thus, applicant's instruction causes the value of an operand to be shifted according to a shift control parameter specified by the instruction, and that shifted value is loaded to bytes of a destination register, as specified by a field of the instruction representing a mask.

Karguth describes a processor that includes a register file of multiple registers, each of which is connected to an input of a plurality of multiplexers, where each multiplexer is associated with a shift/mask circuit which permits the selection of a particular portion of the contents of the register file for use as an operand. With respect to the shift/mask circuit, Karguth explains:

Each of shift/mask units 28 are also controlled by control and instruction decode circuitry 32, to select either a portion or all of the register contents forwarded thereto by its associated operand multiplexer 26, and present this selected portion to ALU 30. In this example, the data presented by each shift/mask unit 28 to ALU 30, and upon which ALU 30 operates in executing the instruction, may be eight bits, sixteen bits, or thirty-two bits, as will be described in further detail hereinbelow. (Col. 7, lines 56-67)

and

As a result of this construction, one or more operands selected by one of operand multiplexers 26 according to the instruction are individually shifted and masked by the associated shift/mask unit 28, in the same machine cycle, prior to presentation to ALU 30 for execution of the instruction. This single-cycle operation is in stark contrast to conventional microprocessor architectures, which necessitate separate shift and mask operations on each operand, necessarily consuming additional machine cycles. Similarly, the writeback operation according to the preferred embodiment of the present invention also shifts the output result from ALU 30 into the proper position within the result word, with the remaining bits (if any) "zero-packed", and presented on write-back bus WBBUS to the destination register selected by destination selector 40 under the control of the instruction bits 7:0 described hereinabove; as noted above, separate write enable control is effected by destination selector 40, so that one or more of the byte locations may be written with the contents of writeback bus WBBUS (the remaining bits being masked from the write). This construction and operation is especially beneficial in packed data structures, where contents of external memory are tightly packed within on-chip memory for high performance operation, as in packed data structure processor 25 according to the preferred embodiment of the present invention. (Col. 10, lines 10-34)

Karguth's shift/mask units select particular portions of the content of the multiplexed registers, and move the selected portions to the ALU for further processing. In other words, the

"shift" functionality of a shift/mask unit refers to moving the content of selected portions. Karguth, however, does not describe that the content of an operand undergoes a shifting operation, and that the shifted content is what's loaded into a destination register. Accordingly, Karguth fails to disclose or suggest at least the feature of "a local register instruction that loads one or more bytes, specified by a multiple-bit field of the instruction, within a local destination register with a shifted value of another operand, the field representing a mask in which each bit of the mask identifies a different byte of the destination register," as required by applicant's independent claim 17.

Furthermore, Karguth describes several instruction formats for use with its processor, including, for example:

OPCODE IO Operand2 SRCsel SRC DSTsel DST
(31:25) (24) (23:16) (15:13) (12:8) (7:5) (4:0)

In this example, the **OPCODE** field in bit positions 31:25 of the instruction specifies the particular arithmetic or logical operation to be carried out (e.g., **ADD**, **XOR**, **SET**, **AND**, etc.). Bit positions 4:0 and 12:8 provide a five-bit selection code by way of which the destination and source registers, respectively, for the instruction are addressed in register file 24 (register file 24 containing thirty-two registers **REG0** through **REG31**). As noted above, register **REG31** cannot serve as a destination.

Bit positions 7:5 and 15:13 each provide a three-bit code, by way of which the desired portion of the destination and source registers are to be selected by shift/mask units 28. In this example, the three bit code, in each case, make this selection of locations within the register as follows (referring also to FIG. 4):

Bits Meaning 000 Select **BYTE0** (bits 7:0) of addressed register 001 Select **BYTE1** (bits 15:8) of addressed register 010 Select **BYTE2** (bits 23:16) of addressed register 011 Select **BYTE3** (bits 31:23) of addressed register 100 Select **WORD0** (bits 15:0) of addressed register 101 Select **WORD1** (bits 23:8) of addressed register 110 Select **WORD2** (bits 31:16) of addressed register 111 Select **DWORD** (bits 31:0) of addressed register

As noted above, the same coding applies to bits 15:13 for the source register as to bits 5:3 for the destination register. As apparent from this coding, any one of the byte, word, or double-word operand lengths are readily selectable for use as the operand in the instruction, with the results writable into any one of the operand lengths in the destination register.

According to the preferred embodiment of the present invention, the second operand in the arithmetic or logical instruction may be an immediate operand, or the contents of one of the registers in register file 24. In this

regard, bit position 24 (IO) in this instruction indicates, when set, that bit positions 23:16 contain an immediate operand value for use in the arithmetic or logical operation specified by the OPCODE portion of the instruction. If bit position (IO) is not set, bit positions 23:16 contain a second source register in register file 24, with bit positions 20:16 selecting one of the thirty-two registers in register file 24 as the source register, and bit positions 23:21 containing a three-bit code indicating the location (BYTE, WORD, or DWORD) within that source register, coded as described above. In this case of the second source register, the appropriate ones of operand multiplexers 26 and shift/mask units 28 again select the addressed portion of the addressed register in register file 24, for application to ALU 30 along with the first source operand determined by bit positions 15:8 as described above. (Col. 9, lines 20-23)

Karguth does not describe that this instruction, or any of Karguth's other instructions, includes a shift control parameter. Karguth does not describe a shift control parameter specified in an instruction according to which the value of an operand specified by the instruction is shifted. Thus, Karguth also fails to disclose or suggest at least the feature of "the shifted value of the other operand being shifted according to a shift control parameter specified by the local register instruction," required by applicant's independent claim 17.

As explained in applicant's previous replies to office action for the above-identified application, none of the other references relied upon by the examiner is believed to remedy the foregoing deficiencies of Karguth.

Because none of the cited references discloses or suggests, alone or in combination, at least the features of "a local register instruction that loads one or more bytes, specified by a multiple-bit field of the instruction, within a local destination register with a shifted value of another operand, the field representing a mask in which each bit of the mask identifies a different byte of the destination register, the shifted value of the other operand being shifted according to a shift control parameter specified by the local register instruction," applicant's independent claim 17, and the claims that depend from it, are therefore patentable over the cited art.

Independent claim 26 describes an apparatus featuring "a command that causes the ALU to load one or more bytes, specified by a multiple-bit field of the command, within a destination register of a selected microengine with a shifted value of another one or more bytes of a source register, the field representing a mask in which each bit of the mask identifies a different byte of

the destination register, the shifted value of the other one or more bytes of the source register being shifted according to a shift control parameter specified by the command." For similar reasons as those provided with respect to independent claim 17, at least these features are not disclosed by the cited art. Independent claim 26 and the claims that depend from it are therefore patentable over the cited art.

The examiner also rejected claims 17, 20-21, 23-24, 26 and 28-29 on the ground of non-statutory obviousness-type double patenting as being unpatentable over claims 1, 3, 9, 10 of U.S. Patent No. 6,668,317 to Bernstein in view of Karguth, and further in view of Probst. The examiner also rejected claims 17-21, 23-26 and 28-29 on the ground of non-statutory obviousness-type double patenting as being unpatentable over claims 1, 2-3 and 5-6 of U.S. Patent No. 7,191,309 to Wolrich, in view of Karguth.

In view of the amendments to applicant's independent claims 17 and 26, and further in view of applicant's arguments in relation to the prior art reference of Karguth as provided herein, applicant respectfully submits that amended claims 17 and 26, and the respective claims depending from them, are patentably distinct over the claims of Bernstein and Wolrich, and should not be rejected on grounds of non-statutory obviousness-type double patenting.

It is believed that all the rejections and/or objections raised by the examiner have been addressed.

In view of the foregoing, applicant respectfully submits that the application is in condition for allowance and such action is respectfully requested at the examiner's earliest convenience.

All of the dependent claims are patentable for at least the reasons for which the claims on which they depend are patentable.

Canceled claims, if any, have been canceled without prejudice or disclaimer.

Any circumstance in which the applicant has (a) addressed certain comments of the examiner does not mean that the applicant concedes other comments of the examiner, (b) made arguments for the patentability of some claims does not mean that there are not other good reasons for patentability of those claims and other claims, or (c) amended or canceled a claim

does not mean that the applicant concedes any of the examiner's positions with respect to that claim or other claims.

No fees are believed due. Please apply any other required fees to deposit account 06-1050, referencing the attorney docket number shown above.

Respectfully submitted,

Date:

June 5, 2007

Ido Rabinovitch

Ido Rabinovitch
Attorney for Intel Corporation
Reg. No. L0080

PTO Customer No. 20985
Fish & Richardson P.C.
Telephone: (617) 542-5070
Facsimile: (617) 542-8906